

# Parametric Bicubic Surfaces

replace  $t$  with  $s$   
 $Q(s) = S \cdot M \cdot G \leftarrow$  parametric cubic curve

$$Q(s,t) = S \cdot M \cdot G(t) = S \cdot M \cdot \begin{bmatrix} G_1(t) \\ G_2(t) \\ G_3(t) \\ G_4(t) \end{bmatrix}$$

and  $G_i(t) = T \cdot M \cdot \bar{G}_i$

where  $\bar{G}_i = \begin{bmatrix} g_{i1} \\ g_{i2} \\ g_{i3} \\ g_{i4} \end{bmatrix}$

$\leftarrow G_1(t), G_2(t), G_3(t), G_4(t)$   
 each is defined by its own set of  $\bar{G}_i$



We use the identity  $(A \cdot B \cdot C)^T = C^T \cdot B^T \cdot A^T$   
 to get:

$$Q(s,t) = S \cdot M \cdot \begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \\ g_{31} & g_{32} & g_{33} & g_{34} \\ g_{41} & g_{42} & g_{43} & g_{44} \end{bmatrix} \cdot M^T \cdot T^T$$

$\uparrow$   
 Cartesian product surface

$0 \leq s, t \leq 1$

Ex: Hermite surface

$$x(s,t) = S \cdot M_H \cdot G_{Hx}(t) = S \cdot M_H \cdot \begin{bmatrix} P_1(t) \\ P_4(t) \\ R_1(t) \\ R_4(t) \end{bmatrix} \cdot X$$

where  $P_{1x}(t) = T \cdot M_H \cdot \begin{bmatrix} g_{11} \\ g_{12} \\ g_{13} \\ g_{14} \end{bmatrix} \cdot X$

$$P_{4x}(t) = T \cdot M_H \cdot \begin{bmatrix} g_{21} \\ g_{22} \\ g_{23} \\ g_{24} \end{bmatrix} \cdot X$$

$$R_{1x}(t) = T \cdot M_H \cdot \begin{bmatrix} g_{31} \\ g_{32} \\ g_{33} \\ g_{34} \end{bmatrix}_x, \quad R_{4x}(t) = T \cdot M_H \cdot \begin{bmatrix} g_{41} \\ g_{42} \\ g_{43} \\ g_{44} \end{bmatrix}_x$$

$$\begin{bmatrix} P_1(t) & P_4(t) & R_1(t) & R_4(t) \end{bmatrix}_x = T \cdot M_H \cdot G_{Hx}^T$$

$$\text{Where } G_{Hx}^T = \begin{bmatrix} g_{11} & g_{21} & g_{31} & g_{41} \\ g_{12} & g_{22} & g_{32} & g_{42} \\ g_{13} & g_{23} & g_{33} & g_{43} \\ g_{14} & g_{24} & g_{34} & g_{44} \end{bmatrix}_x$$

Transposing both sides,

$$\begin{bmatrix} P_1(t) \\ P_4(t) \\ R_1(t) \\ R_4(t) \end{bmatrix}_x = \begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \\ g_{31} & g_{32} & g_{33} & g_{34} \\ g_{41} & g_{42} & g_{43} & g_{44} \end{bmatrix} \cdot M_H \cdot T^T = G_{Hx} \cdot M_H^T \cdot T^T$$

$\therefore$

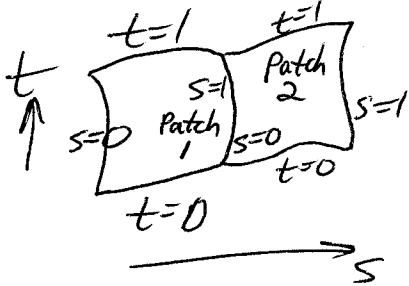
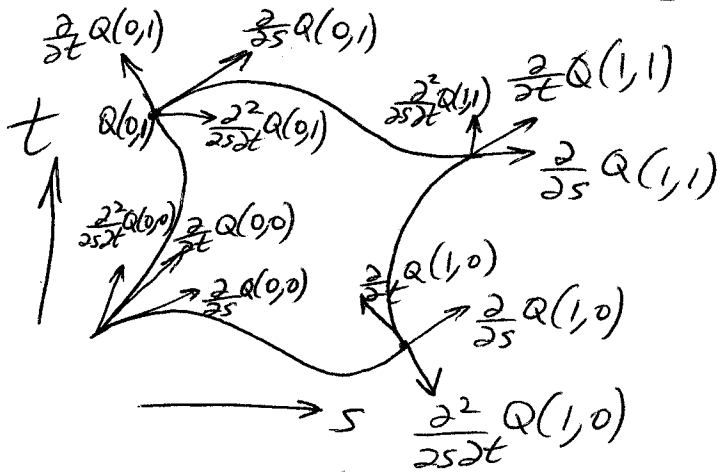
$$x(s, t) = S \cdot M_H \cdot G_{Hx} \cdot M_H^T \cdot T^T$$

$$G_{4 \times 4} = \begin{bmatrix} x(0,0) & x(0,1) & \frac{\partial}{\partial t} x(0,0) & \frac{\partial}{\partial t} x(0,1) \\ x(1,0) & x(1,1) & \frac{\partial}{\partial t} x(1,0) & \frac{\partial}{\partial t} x(1,1) \\ \frac{\partial^2}{\partial s^2} x(0,0) & \frac{\partial^2}{\partial s^2} x(0,1) & \frac{\partial^2}{\partial s \partial t} x(0,0) & \frac{\partial^2}{\partial s \partial t} x(0,1) \\ \frac{\partial^2}{\partial s^2} x(1,0) & \frac{\partial^2}{\partial s^2} x(1,1) & \frac{\partial^2}{\partial s \partial t} x(1,0) & \frac{\partial^2}{\partial s \partial t} x(1,1) \end{bmatrix}$$

← starting pts for  $P_{1 \times 4}(t)$   
 ← "  $P_{4 \times 4}(t)$   
 ← "  $R_{1 \times 4}(t)$   
 ← "  $R_{4 \times 4}(t)$

the twist vectors

If all 4 twist vectors are 0  $\Rightarrow$  Ferguson surface



Patch 1

$$\begin{bmatrix} \overline{g_{21}} & \overline{g_{22}} & \overline{g_{23}} & \overline{g_{24}} \\ \overline{g_{41}} & \overline{g_{42}} & \overline{g_{43}} & \overline{g_{44}} \end{bmatrix}$$

Patch 2

$$\begin{bmatrix} g_{21} & g_{22} & g_{23} & g_{24} \\ kg_{41} & kg_{42} & kg_{43} & kg_{44} \end{bmatrix}$$

For  $G^1$  continuity  
If  $k=1$ ,  $C^1$  "

$k > 0$

for  $C^0$ , only control points must be shared along edge

Consider first the simpler problem of constructing a surface patch given only two of its boundaries,  $\mathbf{r}(0,v)$  and  $\mathbf{r}(1,v)$ . If we use linear interpolation in the  $u$ -direction, we obtain the ruled surface

$$\mathbf{r}_1(u,v) = (1 - u)\mathbf{r}(0,v) + u\mathbf{r}(1,v) \quad (7.1)$$

Alternatively, linear interpolation in the  $v$ -direction gives a surface which fits the other two boundaries:

$$\mathbf{r}_2(u,v) = (1 - v)\mathbf{r}(u,0) + v\mathbf{r}(u,1) \quad (7.2)$$

These two surfaces are sketched in Figure 7.2 below.

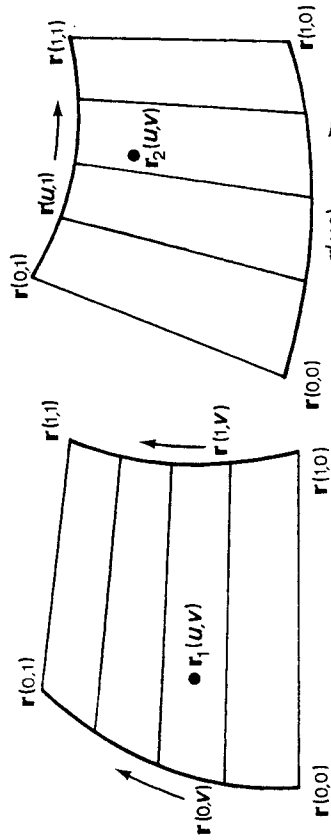


Figure 7.2 - Surface Patches Resulting from Linear Interpolation between Pairs of Boundaries

Now the sum  $\mathbf{r}_1 + \mathbf{r}_2$  represents a surface patch each of whose boundaries is the sum of the desired boundary curve with the linear interpolant between the end-points of that curve. For instance, it is easily verified from (7.1) and (7.2) that the boundary corresponding to  $v = 0$  is not  $\mathbf{r}(u,0)$  but

$$\mathbf{r}(u,0) + [(1 - u)\mathbf{r}(0,0) + u\mathbf{r}(1,0)]$$

Then if we can find a surface patch  $\mathbf{r}_3(u,v)$  whose boundaries are the unwanted linear interpolants, we can recover the original boundary curves by forming  $\mathbf{r}_1 + \mathbf{r}_2 - \mathbf{r}_3$ . It is easy to construct  $\mathbf{r}_3$ ; its  $v = 0$  and  $v = 1$  boundaries must be

$$(1 - u)\mathbf{r}(0,0) + u\mathbf{r}(1,0), \quad (1 - u)\mathbf{r}(0,1) + u\mathbf{r}(1,1)$$

respectively, and a further linear interpolation in the  $v$ -direction then gives

$$\begin{aligned} \mathbf{r}_3(u,v) = & (1 - u)(1 - v)\mathbf{r}(0,0) + u(1 - v)\mathbf{r}(1,0) \\ & + (1 - u)v\mathbf{r}(0,1) + uv\mathbf{r}(1,1). \end{aligned} \quad (7.3)$$

## Chapter 7

# Composite surfaces

### 7.1 INTRODUCTION: COONS PATCHES

In this chapter we discuss a number of methods for defining surfaces. It was explained in Section 6.1 that some of these methods are based on a framework built up of two intersecting families of curves, and we start by considering procedures of this type. It will be assumed that the necessary mesh of curves has already been constructed using an appropriate method from Chapter 6. For greater generality we will work initially in terms of parametric curves, distinguishing fore-and-aft curves from transverse curves by defining them in terms of the parameters  $u$  and  $v$  respectively. We will later specialise our results to the nonparametric case.

The network of curves divides the surface into an assembly of topologically rectangular patches, each of which has as its boundaries two  $u$ -curves and two  $v$ -curves, as shown in Figure 7.1. Here it is assumed that  $u$  and  $v$  run from 0 to 1 along the relevant boundaries; then  $\mathbf{r}(u,v)$ ,  $0 < u, v < 1$ , represents the interior of the surface patch, while  $\mathbf{r}(u,0)$ ,  $\mathbf{r}(1,v)$ ,  $\mathbf{r}(u,1)$  and  $\mathbf{r}(0,v)$  represent the four known boundary curves. The problem of defining a surface patch, then, is that of finding a suitably well-behaved function  $\mathbf{r}(u,v)$  which reduces to the correct boundary curve when  $u = 0, u = 1, v = 0$  or  $v = 1$ .

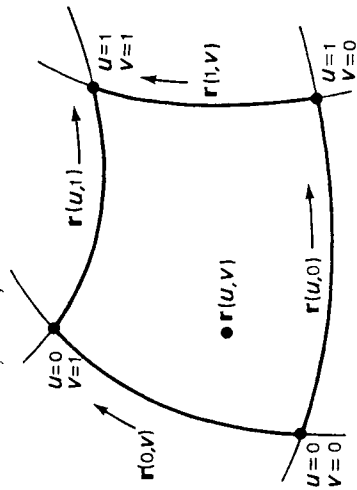


Figure 7.1. - A Parametric Surface Patch

14/1

As in the derivation of (7.5), we find that  $r_1 + r_2$  does not give us the surface we require. To get this, we have to subtract the further surface  $r_3(u, v)$  which results when we apply the same interpolation technique in both directions, using corner data alone. The resulting equation is:

$$r(u, v) = [\alpha_0(u) \alpha_1(u) \beta_0(u) \beta_1(u)] \begin{bmatrix} r(0, v) \\ r(1, v) \\ r_u(0, v) \\ r_u(1, v) \end{bmatrix} + [r(u, 0) \ r(u, 1) \ r_v(u, 0) \ r_v(u, 1)] \begin{bmatrix} \alpha_0(v) \\ \alpha_1(v) \\ \beta_0(v) \\ \beta_1(v) \end{bmatrix} - [\alpha_0(u) \ \alpha_1(u) \ \beta_0(u) \ \beta_1(u)] \begin{bmatrix} r(0, 0) & r(0, 1) & r_v(0, 0) & r_v(0, 1) \\ r(1, 0) & r(1, 1) & r_v(1, 0) & r_v(1, 1) \\ r_u(0, 0) & r_u(0, 1) & r_{uv}(0, 0) & r_{uv}(0, 1) \\ r_u(1, 0) & r_u(1, 1) & r_{uv}(1, 0) & r_{uv}(1, 1) \end{bmatrix} \begin{bmatrix} \alpha_0(v) \\ \alpha_1(v) \\ \beta_0(v) \\ \beta_1(v) \end{bmatrix} \quad (7.12)$$

Here the three terms on the right are respectively  $r_1$ ,  $r_2$  and  $r_3$  in the foregoing argument. Note that corner values of the cross-derivative  $r_{uv}$  are necessary for the construction of  $r_3$ . The reader may find it helpful to check in detail that, provided the blending functions satisfy (7.7) to (7.10), the surface represented by (7.12) has the specified boundary curves and cross-boundary gradients. Using patches of this kind we can match surface normal direction across boundaries and construct a composite surface which is everywhere smooth.

Working along similar lines, we may derive the equation of a patch with specified cross-boundary second derivatives. Two more blending functions are needed, and the square matrix involved will be of order  $6 \times 6$ . Most practical surface-defining systems avoid this degree of complexity, however.

The cross-derivative  $r_{uv}$  which occurs in the patch equation (7.12) appears frequently in this chapter, and a brief discussion of its physical significance may be useful. Let us first consider a surface  $z = z(x, y)$  in Cartesian coordinates.

Since  $z_{xy} = \frac{\partial}{\partial x} \left( \frac{\partial z}{\partial y} \right)$ , we see that the mixed or cross-derivative measures the rate of change in the  $x$ -direction of the slope of the surface in the  $y$ -direction. Then  $z_{xy}$  is a measure of *twist* in the surface. If  $z_{xy}$  is a continuous function of  $x$  and  $y$ , it may be shown analytically that  $z_{yx} = z_{xy}$ .

By analogy with the Cartesian case, the parametric cross-derivative  $r_{uv}$  is often called a **twist vector**. Although we follow custom and use this term ourselves in what follows, we must point out that it can be very misleading. This is

because the value of  $r_{uv}$  at a point on a parametric surface depends not so much on the geometrical properties of the surface as on the way in which it is parametrised. A simple example serves to show this. Suppose that  $a$ ,  $b$  and  $c$  are constant vectors such that  $b \times c \neq 0$ , and let  $u$ ,  $v$  be any scalars. Then

$$r(u, v) = a + ub + vc$$

defines a flat plane containing the point whose position vector is  $a$ , and parallel to the vectors  $b$  and  $c$ . A flat plane is not a twisted surface, and it is therefore reassuring to find that  $r_{uv} = 0$ . But now consider

$$r(u, v) = a + ub + wvc$$

This reparametrisation of the previous equation defines the same flat plane, but now  $r_{uv} = c \neq 0$ . We conclude that caution must be exercised in interpreting the 'twist vector' in geometrical terms, since  $r_{uv} \neq 0$  does not necessarily imply a twist in a surface.

### 7.2 TENSOR-PRODUCT SURFACES

It is possible to simplify the Coons patch equation (7.12) considerably by suitably defining the boundary curves and cross-boundary gradients. Using a set of blending functions which satisfy the conditions (7.7) to (7.10), we may express a curve segment in terms of its end points and end tangents by

$$r(u) = \alpha_0(u)r(0) + \alpha_1(u)r(1) + \beta_0(u)r'(0) + \beta_1(u)r'(1) \quad (7.13)$$

This formulation permits our boundary curves and cross-boundary gradients to be written as

$$r(i, v) = \alpha_0(v)r(i, 0) + \alpha_1(v)r(i, 1) + \beta_0(v)r_v(i, 0) + \beta_1(v)r_v(i, 1),$$

$$r_u(i, v) = \alpha_0(v)r_u(i, 0) + \alpha_1(v)r_u(i, 1) + \beta_0(v)r_{uv}(i, 0) + \beta_1(v)r_{uv}(i, 1), \quad (7.14)$$

where  $i = 0, 1$ , and

$$r(u, j) = \alpha_0(u)r(0, j) + \alpha_1(u)r(1, j) + \beta_0(u)r_u(0, j) + \beta_1(u)r_u(1, j),$$

$$r_v(u, j) = \alpha_0(u)r_v(0, j) + \alpha_1(u)r_v(1, j) + \beta_0(u)r_{uv}(0, j) + \beta_1(u)r_{uv}(1, j), \quad (7.15)$$

where  $j = 0, 1$ . When these forms are substituted into (7.12) it is found that all three terms are now identical but for the negative sign of the third. The equation therefore reduces to

The surface  $\mathbf{r} = \mathbf{r}_1 + \mathbf{r}_2 + \mathbf{r}_3$  obtained from (7.1), (7.2) and (7.3) is conveniently expressed in the matrix form

$$\mathbf{r}(u, \nu) = \begin{bmatrix} (1-u) \\ \nu \end{bmatrix} \begin{bmatrix} \mathbf{r}(0, \nu) \\ \mathbf{r}(1, \nu) \end{bmatrix} + \begin{bmatrix} \mathbf{r}(u, 0) \\ \mathbf{r}(u, 1) \end{bmatrix} \begin{bmatrix} 1-\nu \\ \nu \end{bmatrix} - \begin{bmatrix} (1-u) \\ \nu \end{bmatrix} \begin{bmatrix} \mathbf{r}(0, 1) \\ \mathbf{r}(1, 1) \end{bmatrix} \begin{bmatrix} 1-\nu \\ \nu \end{bmatrix}. \quad (7.4)$$

Successive substitution of  $u = 0, u = 1, \nu = 0$  and  $\nu = 1$  quickly confirms that the patch defined by (7.4) has the four original curves as its boundaries.

This patch, constructed solely in terms of information given on its boundary and certain auxiliary scalar functions of  $u$  and  $\nu$ , is the most elementary of a class of surfaces originally studied by Coons (1967), and which have since become known as **Coons patches**. The treatment given here is essentially that of Forrest (1972a). The auxiliary functions  $u, (1-u), \nu$  and  $(1-\nu)$  are called **blending functions**, because their effect is to blend together four separate boundary curves to give a single well-defined surface.

An important generalisation of (7.4) may be made immediately. The linear blending functions in the patch equation result from the use of *uniform* linear interpolation. In fact, we may replace  $(1-u), u$  by  $\alpha_0(u), \alpha_1(u)$  where the blending functions are now any functions such that  $1 - \alpha_1 = \alpha_0$  and

$$\begin{aligned} \alpha_0(0) &= 1, & \alpha_0(1) &= 0 \\ \alpha_1(0) &= 0, & \alpha_1(1) &= 1. \end{aligned}$$

With this change, the  $u$ -interpolations performed in the derivation of (7.4) remain linear, but the rate of motion of  $\mathbf{r}$  along the line of interpolation is now no longer constant as  $u$  increases uniformly from 0 to 1. The corresponding replacement of  $(1-\nu), \nu$  by  $\alpha_0(\nu), \alpha_1(\nu)$  gives

$$\mathbf{r}(u, \nu) = \begin{bmatrix} \alpha_0(u) \\ \alpha_1(u) \end{bmatrix} \begin{bmatrix} \mathbf{r}(0, \nu) \\ \mathbf{r}(1, \nu) \end{bmatrix} + \begin{bmatrix} \mathbf{r}(u, 0) \\ \mathbf{r}(u, 1) \end{bmatrix} \begin{bmatrix} \alpha_0(\nu) \\ \alpha_1(\nu) \end{bmatrix} - \begin{bmatrix} \alpha_0(u) \\ \alpha_1(u) \end{bmatrix} \begin{bmatrix} \mathbf{r}(0, 1) \\ \mathbf{r}(1, 1) \end{bmatrix} \begin{bmatrix} \alpha_0(\nu) \\ \alpha_1(\nu) \end{bmatrix}. \quad (7.5)$$

The blending functions  $\alpha_0$  and  $\alpha_1$  are customarily chosen to be continuous and

monotone (in other words continuously increasing or continuously decreasing) over the interval  $0 \leq u \leq 1$ . In practice polynomial blending functions are generally used, for both analytical and computational convenience. The condition that  $1 - \alpha_1 = \alpha_0$  may be dispensed with, provided  $\alpha_0$  and  $\alpha_1$  satisfy the other stated requirements. In this case, however, the interpolations between opposite boundaries corresponding to equations (7.1) and (7.2) are nonlinear. In consequence the patch represented by (7.5) does not then reduce to a plane when all its boundaries are coplanar.

Given a network of curves, then, we can construct a composite surface made up of patches of the type described. This surface will have only positional continuity across patch boundaries, however. The gradient continuity essential for most practical applications may be achieved by using a less elementary kind of patch defined not only in terms of its boundary curves but also in terms of its cross-boundary slopes  $\mathbf{r}_u(u, 0), \mathbf{r}_u(u, 1), \mathbf{r}_\nu(u, 0)$  and  $\mathbf{r}_\nu(u, 1)$ .<sup>†</sup>

The equation for such a patch may be derived in a similar manner to (7.5), except that we now use generalised Hermite interpolation rather than generalised linear interpolation. The surface

$$\mathbf{r}_1(u, \nu) = \alpha_0(u)\mathbf{r}(0, \nu) + \alpha_1(u)\mathbf{r}(1, \nu) + \beta_0(u)\mathbf{r}_u(0, \nu) + \beta_1(u)\mathbf{r}_u(1, \nu) \quad (7.6)$$

interpolates the two boundaries  $\mathbf{r}(0, \nu)$  and  $\mathbf{r}(1, \nu)$  and gives the specified slopes across these boundaries if the four blending functions satisfy

$$\begin{aligned} \alpha_0(0) &= 1, & \alpha_0(1) &= 0, \\ \alpha_1(0) &= 0, & \alpha_1(1) &= 1, \end{aligned} \quad (7.7)$$

$$\alpha_0'(0) = \alpha_0'(1) = \alpha_1'(0) = \alpha_1'(1) = 0, \quad (7.8)$$

$$\beta_0(0) = \beta_0(1) = \beta_1(0) = \beta_1(1) = 0 \quad (7.9)$$

$$\text{and} \quad \beta_0'(0) = 1, \quad \beta_0'(1) = 0, \quad (7.10)$$

$$\beta_1'(0) = 0, \quad \beta_1'(1) = 1.$$

The reader may easily verify this for himself by setting  $u = 0$  and  $u = 1$  in (7.6) and in (7.6) differentiated partially with respect to  $u$ . Similarly the two curves  $\mathbf{r}(u, 0)$  and  $\mathbf{r}(u, 1)$ , with their associated gradient data, are interpolated by

$$\mathbf{r}_2(u, \nu) = \alpha_0(\nu)\mathbf{r}(u, 0) + \alpha_1(\nu)\mathbf{r}(u, 1) + \beta_0(\nu)\mathbf{r}_\nu(u, 0) + \beta_1(\nu)\mathbf{r}_\nu(u, 1). \quad (7.11)$$

<sup>†</sup>It is possible to ensure automatic continuity of cross-boundary gradient between elementary patches by suitably choosing the blending functions. But this allows less flexibility, and the resulting surfaces often exhibit unwanted flat regions (see Forrest's Appendix 2 in Bézier, 1972).